

5/5/1 (Item 1 from file: 347)  
DIALOG(R) File 347:JAPIO  
(c) 2000 JPO & JAPIO. All rts. reserv.

04332694 \*\*Image available\*\*  
DEBUGGING DEVICE FOR DISTRIBUTED PROGRAM

PUB. NO.: 05-324394 JP 5324394 A]  
PUBLISHED: December 07, 1993 (19931207)  
INVENTOR(s): KAMIMURA MARI  
TAMURA HIDEO  
APPLICANT(s): NEC CORP [000423] (A Japanese Company or Corporation), JP  
(Japan)  
NEC SOFTWARE LTD [491061] (A Japanese Company or Corporation)  
, JP (Japan)  
APPL. NO.: 04-004373 [JP 924373]  
FILED: January 14, 1992 (19920114)  
INTL CLASS: [5] G06F-011/28; G06F-015/16  
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units);  
45.4 (INFORMATION PROCESSING -- Computer Applications)  
JOURNAL: Section: P, Section No. 1709, Vol. 18, No. 153, Pg. 64, March  
14, 1994 (19940314)

#### ABSTRACT

PURPOSE: To provide the debugging device for the distributed program capable of reducing the burden of an operator by a system operation procedure and a debugging command independent of a computer system being a single computer system.

CONSTITUTION: This device is provided with application programs 8 and 14, a presence information input part 1 for indicating correspondence to the computer systems 100 and 200, a debugging command input part 4 in an expression form independent of the computer system, a host control part 5 and a server control part 11 for retrieving the computer system during the execution of the application program of debugging command information and collecting the debugged result information of the expression form independent of the computer system, debugging command form conversion parts 6 and 12 for converting transferred debugging command information to the expression form of the computer system, debugging parts 7 and 13 for obtaining the debugged result with the converted debugging command information and debugged result form conversion parts 9 and 15 for displaying the debugged result in the expression form independent of the computer system.

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平5-324394

(43)公開日 平成5年(1993)12月7日

(51)Int.Cl.<sup>5</sup>

G 0 6 F 11/28  
15/16

識別記号

庁内整理番号

J 9290-5B  
4 5 0 Z 9190-5L

F I

技術表示箇所

審査請求 未請求 請求項の数1(全 5 頁)

(21)出願番号 特願平4-4373

(22)出願日 平成4年(1992)1月14日

(71)出願人 000004237

日本電気株式会社  
東京都港区芝五丁目7番1号

(71)出願人 000232092

日本電気ソフトウェア株式会社  
東京都港区高輪2丁目17番11号

(72)発明者 上村 万里

東京都港区芝五丁目7番1号日本電気株式  
会社内

(72)発明者 田村 秀雄

東京都港区高輪二丁目17番11号日本電気ソ  
フトウェア株式会社内

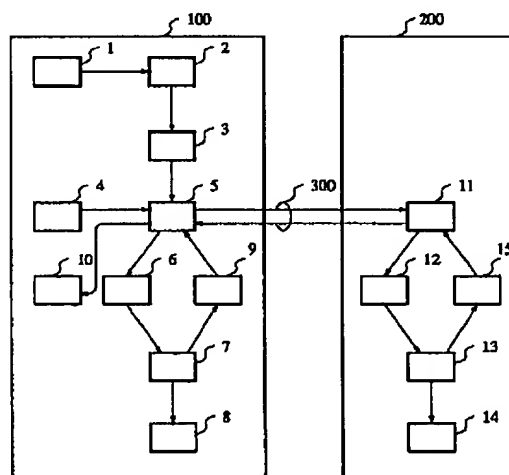
(74)代理人 弁理士 京本 直樹 (外2名)

(54)【発明の名称】 分散プログラムのデバッグ装置

(57)【要約】 (修正有)

【目的】単一の計算機システムで計算機システムに依存しないデバッグコマンドとシステム操作手順で操作員の負荷を軽減できる分散プログラムのデバッグ装置の提供。

【構成】アプリケーションプログラム8および14と、計算機システム100および200との対応を示す存在情報入力部1と、計算機システムに依存しない表現形式のデバッグコマンド入力部4と、デバッグコマンド情報のアプリケーションプログラムを実行中の計算機システムを検索しその計算機システムに依存しない表現形式のデバッグ結果情報を収集するホスト制御部5とサーバ制御部11と、転送デバッグコマンド情報をその計算機システムの表現形式に変換するデバッグコマンド形式変換部6および12と、変換されたデバッグコマンド情報でデバッグ結果を得るデバッグ部7および13と、デバッグ結果を計算機システムに依存しない表現形式の表示用デバッグ結果形式変換部9および15とを含む。



100,200: 計算機システム  
300: 通信路  
1: 存在情報入力部  
2: 存在情報登録部  
3: 存在情報テーブル  
4: デバッグコマンド入力部  
5: ホスト制御部  
6,12: デバッグコマンド形式変換部  
7,13: デバッグ部  
8,14: アプリケーションプログラム  
9,15: デバッグ結果形式変換部  
10: デバッグ結果出力部  
11: サーバ制御部

## 【特許請求の範囲】

【請求項 1】 アプリケーションプログラムとこれを実行する計算機システムとの対応を示す存在情報を入力する存在情報入力手段と、前記入力された存在情報を登録しこれを保持する存在情報登録保持手段と、前記計算機システムに依存しない表現形式でデバッグコマンド情報を入力するデバッグコマンド入力手段と、前記登録保持されている存在情報を参照して前記入力されたデバッグコマンド情報の対象となっているアプリケーションプログラムを実行している計算機システムを検索しこの計算機システムに前記入力されたデバッグコマンド情報を転送しその計算機システムに依存しない表現形式で表現されたデバッグ結果情報を収集する転送収集手段と、前記転送されたデバッグコマンド情報を転送された計算機システムに依存する表現形式に変換するデバッグコマンド形式変換手段と、前記変換されたデバッグコマンド情報により前記アプリケーションプログラムにデバッグのための操作を行ないそのデバッグ結果を得るデバッグ手段と、前記デバッグ結果を計算機システムに依存しない表現形式で表示するためのデバッグ結果形式変換手段と、前記収集されたデバッグ結果を出力するデバッグ結果出力手段とを含むことを特徴とする分散プログラムのデバッグ装置。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 本発明は分散プログラムのデバッグ装置に関し、特に複数の計算機システムにおける分散プログラムのデバッグ装置に関する。

## 【0002】

【従来の技術】 複数の計算機システム上にそれぞれ実装され、相互に関連しあって、ひとつのまとまった機能を実現するアプリケーションプログラム群の開発作業において、デバッグをおこなう従来の分散プログラムのデバッグ装置の場合、個々の計算機システムに操作員を配置し、互いに同期を取ってデバッグのための指示を投入する必要があった。

【0003】 また、個々の計算機システムにおけるデバッグ装置のコマンドの形式は、通常互いに異なるので、アプリケーションプログラム群を実行するすべての計算機システムのコマンドの形式を習熟している必要があった。

【0004】 同様に、計算機システムごとのシステム操作手順も、通常互いに異なるので、アプリケーションプログラム群を実行するすべての計算機システムのシステム操作手順も操作員は習熟している必要があった。

【0005】 さらに、デバッグ作業において、個々のアプリケーションプログラムをどの計算機システム上で実行しているかを操作員に予め伝えておく必要があり、操作員は、現在デバッグ作業の対象としている個々のアプリケーションプログラムと、アプリケーションプログラ

ムを実行している計算機システムとの対応を常に念頭において、コマンドを投入する必要があった。

## 【0006】

【発明が解決しようとする課題】 上述した従来の分散プログラムのデバッグ装置は、相互に関連しあってひとつのまとまった機能を実現するアプリケーションプログラム群のそれぞれを実装している個々の計算機システムは遠隔地に配置する場合が多く、かつ、互いに同期をとりながらデバッグ作業を行なう必要があるため、ひとりの操作員で作業を行なうのは通常不可能である。また、個々の計算機ごとに異なるコマンドの形式やシステム操作手順をひとりの操作員ですべて理解するのも、計算機システムの数が増えるに従い困難となる。

【0007】 複数の操作員により、分担してデバッグ作業を行なうように作業の手順を変更したとしても、個々の計算機システムのコマンドの形式やシステム操作手順をすべて網羅するだけの操作員を確保するのは計算機システムの数が増えるに従い困難となり、確保できたとしても、デバッグ作業の同期をとるために要する工数が膨大になり現実的でない。

【0008】 さらに、操作員が常にアプリケーションプログラムと、アプリケーションプログラムを実行する計算機システムとの対応を覚えていなければならないため、操作員の負荷が高くなるという問題点がある。

【0009】 本発明の目的は、複数の計算機システムに実装され全体で単一の機能を実現するアプリケーションプログラム群を、単一の計算機システムより計算機システムに依存しないデバッグコマンドおよびシステム操作手順でデバッグでき操作員の負荷を大幅に軽減できる分散プログラムのデバッグ装置を提供することにある。

## 【0010】

【課題を解決するための手段】 本発明の分散プログラムのデバッグ装置は、アプリケーションプログラムとこれを実行する計算機システムとの対応を示す存在情報を入力する存在情報入力手段と、前記入力された存在情報を登録しこれを保持する存在情報登録保持手段と、前記計算機システムに依存しない表現形式でデバッグコマンド情報を入力するデバッグコマンド入力手段と、前記登録保持されている存在情報を参照して前記入力されたデバッグコマンド情報の対象となっているアプリケーションプログラムを実行している計算機システムを検索しこの計算機システムに前記入力されたデバッグコマンド情報を転送しその計算機システムに依存しない表現形式で表現されたデバッグ結果情報を収集する転送収集手段と、前記転送されたデバッグコマンド情報を転送された計算機システムに依存する表現形式に変換するデバッグコマンド形式変換手段と、前記変換されたデバッグコマンド情報により前記アプリケーションプログラムにデバッグのための操作を行ないそのデバッグ結果を得るデバッグ手段と、前記デバッグ結果を計算機システムに依存しな

## 3

い表現形式で表示するためのデバッグ結果形式変換手段と、前記収集されたデバッグ結果を出力するデバッグ結果出力手段とを含んで構成されている。

【0011】

【実施例】次に、本発明の実施例について図面を参照して説明する。

【0012】図1は本発明の分散プログラムのデバッグ装置の一実施例を示すブロック図である。

【0013】本実施例の分散プログラムのデバッグ装置は、図1に示すように、計算機システム100と200の2システムが通信路300を介して接続されている場合を示している。

【0014】アプリケーションプログラム8と14とはそれぞれ計算機システム100と200とに分散して実装されており、これらのアプリケーションプログラムは2つ揃うことにより1つの機能を実現する。

【0015】最初に起動するアプリケーションプログラムは、アプリケーションプログラム8であり、アプリケーションプログラム14の起動は、予め計算機システム100と200との間で取り決めてある方法により、アプリケーションプログラム8が行なう。

【0016】アプリケーションプログラム14を実行している間、アプリケーションプログラム8は実行を一時停止した状態となっている。一時停止した状態にあるアプリケーションプログラム8は、アプリケーションプログラム14の終了の通知を、予め計算機システム100と200との間で取り決めてある方法により、アプリケーションプログラム14から受け取ると、実行を再開する。

【0017】計算機システム100内のデバッグ部7と計算機システム200内のデバッグ部13は、デバッグ作業の指示を受け取るために、表現形式が異なるデバッグコマンド情報の体系を有している。ただし、操作員は、表現形式がデバッグ部7およびデバッグ部13に依存しない共通形式のデバッグコマンドを用いて、アプリケーションプログラム8およびアプリケーションプログラム14のデバッグ作業を行なう。

【0018】同様に、デバッグ作業の結果を示すデバッグ結果情報の体系も、計算機システム100内のデバッグ部7の有する表現形式、計算機システム200内のデバッグ部13の有する表現形式、およびデバッグ部7およびデバッグ部13に依存しない共通形式なる表現形式、の3つの体系がある。

【0019】操作員は、デバッグ部7およびデバッグ部13に依存しない共通形式のデバッグ結果情報を用いて、アプリケーションプログラム8およびアプリケーションプログラム14のデバッグ作業を行なう。

【0020】次に図1を参照して本実施例の動作について説明する。

【0021】デバッグ作業に先立ち、操作員は存在情報

## 4

入力部1を使用して、アプリケーションプログラム8および14と計算機システム100および200との対応を示す存在情報を存在情報登録部2に入力する。

【0022】存在情報登録部2は、存在情報入力部1より入力されたアプリケーション存在情報を存在情報テーブル3に登録する。

【0023】存在情報テーブル3は、その内部に、存在情報登録部2が登録したアプリケーション存在情報を保持するテーブルである。

10 【0024】アプリケーションプログラム8および14のデバッグ作業を行なうとき、操作員は、計算機システム100内のデバッグ部7および計算機システム200内のデバッグ部13に依存しない共通形式のデバッグコマンド情報を、デバッグコマンド入力部4より入力する。

【0025】デバッグコマンド入力部4は、入力された共通形式のデバッグコマンド情報をホスト制御部5へ出力する。

20 【0026】ホスト制御部5は、存在情報テーブル3に保持されているアプリケーションプログラム存在情報と、デバッグコマンド入力部4より出力された共通形式のデバッグコマンド情報とを比較し、共通形式のデバッグコマンド情報が、計算機システム100内のアプリケーションプログラム8、計算機システム200内のアプリケーションプログラム14、のいずれに対応しているかを判断する。

30 【0027】ホスト制御部5により、共通形式のデバッグコマンド情報が計算機システム100内のアプリケーションプログラム8に対応している、と判断された場合、ホスト制御部5は、デバッグコマンド入力部4より出力された共通形式のデバッグコマンド情報を、計算機システム100内のデバッグコマンド形式変換部6へ出力する。

【0028】デバッグコマンド形式変換部6は、ホスト制御部5より出力された共通形式のデバッグコマンド情報の形式を、計算機システム100内のデバッグ部7の有する体系に属する形式に変換した後、変換した後の情報をデバッグ部7へ出力する。

40 【0029】デバッグ部7は、デバッグコマンド形式変換部6より出力された計算機システム100のデバッグコマンド情報に対応するアプリケーションプログラム8に対するデバッグ操作を行ない、デバッグ操作を行なった結果を、デバッグ結果形式変換部9に出力するデバッグ結果形式変換部9は、デバッグ部7より出力された計算機システム100のデバッグ結果情報の形式を、デバッグ部7、13に依存しない共通形式の体系に属する形式に変換した後、変換した後の情報をホスト制御部5へ出力する。

50 【0030】ホスト制御部5は、デバッグ結果形式変換部9より出力された共通形式のデバッグ結果情報を、デ

## 5

バッグ結果出力部 1 0 へ出力する。

【0 0 3 1】デバッグ結果出力部 1 0 は、ホスト制御部 5 より出力された共通形式のデバッグ結果情報の内容を操作員に伝える。

【0 0 3 2】他方、ホスト制御部 5 により、共通形式のデバッグコマンド情報が計算機システム 2 0 0 内のアプリケーションプログラム 1 4 に対応している、と判断された場合、ホスト制御部 5 は、デバッグコマンド入力部 4 より出力された共通形式のデバッグコマンド情報を、通信路 3 を介して、計算機システム 2 0 0 内のサーバ制御部 1 1 へ出力する。

【0 0 3 3】サーバ制御部 1 1 は、通信路 3 を介してホスト制御部 5 より出力された共通形式のデバッグコマンド情報を、デバッグコマンド形式変換部 1 2 に出力する。

【0 0 3 4】デバッグコマンド形式変換部 1 2 は、サーバ制御部 1 1 より出力された共通形式のデバッグコマンド情報の形式を、計算機システム 2 0 0 内のデバッグ部 1 3 の有する体系に属する形式に変換した後、変換した後の情報をデバッグ部 1 3 へ出力する。

【0 0 3 5】デバッグ部 1 3 は、デバッグコマンド形式変換部 1 2 より出力された計算機システム 2 0 0 のデバッグコマンド情報に対応するアプリケーションプログラム 1 4 に対するデバッグ操作を行ない、デバッグ操作を行なった結果を、デバッグ結果形式変換部 1 5 に出力する。

【0 0 3 6】デバッグ結果形式変換部 1 5 は、デバッグ部 1 3 より出力された計算機システム 2 0 0 のデバッグ結果情報の形式をデバッグ部 7、1 3 に依存しない共通形式の体系に属する形式に変換した後、変換した後の情報

をサーバ制御部 1 1 へ出力する。

【0 0 3 7】サーバ制御部 1 1 は、デバッグ結果形式変換部 1 5 より出力された共通形式のデバッグコマンド情報を、通信路 3 0 0 を介して、ホスト制御部 5 に出力する。

【0 0 3 8】通信路 3 0 0 を介して、サーバ制御部 1 1

## 6

より出力された共通形式のデバッグ結果情報をホスト制御部 5 へ入力した後のホスト制御部 5 およびデバッグ結果出力部 1 0 の動作は、計算機システム 1 0 0 での共通形式のデバッグ結果情報を入力した場合と同一である。

【0 0 3 9】以上の動作の説明より、計算機システム 1 0 0 より、計算機システム 2 0 0 内のアプリケーションプログラム 1 4 のデバッグ作業もできることとなる。

【0 0 4 0】

【発明の効果】以上説明したように、本発明の分散プログラムのデバッグ装置は、1 台の計算機システムに一人の操作員を置くだけで、アプリケーションプログラム群を実行しているすべての計算機システム上のアプリケーションプログラムに対するデバッグ作業を行なうことができ、操作員は、デバッグ作業に関し、統一された 1 種類のコマンド形式の体系やシステム操作手順を覚えるだけでよく、かつ、操作員はアプリケーションプログラムと、アプリケーションプログラムを実行する計算機システムとの対応を覚える必要がないので、操作員に対する負荷を大幅に軽減できるという効果がある。

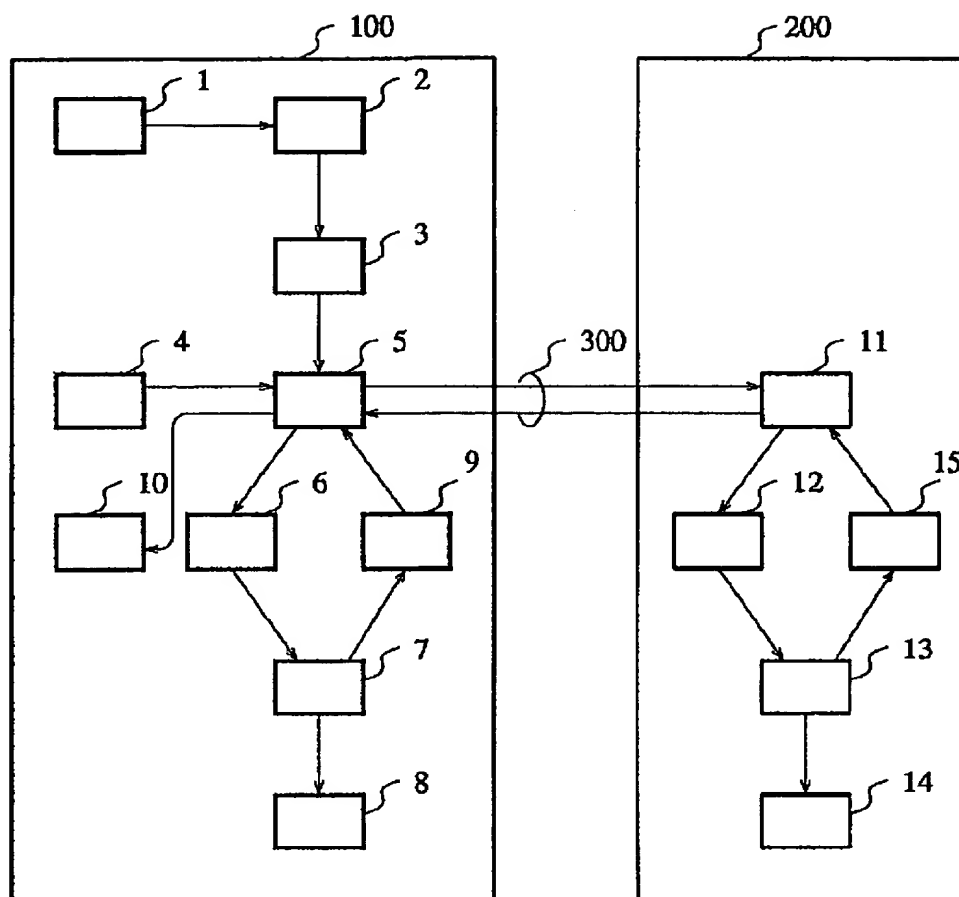
【図面の簡単な説明】

【図 1】本発明の分散プログラムのデバッグ装置の一実施例を示すブロック図である。

【符号の説明】

- 1 存在情報入力部
- 2 存在情報登録部
- 3 存在情報テーブル
- 4 デバッグコマンド入力部
- 5 ホスト制御部
- 6、1 2 デバッグコマンド形式変換部
- 7、1 3 デバッグ部
- 8、1 4 アプリケーションプログラム
- 9、1 5 デバッグ結果形式変換部
- 1 0 デバッグ結果出力部
- 1 1 サーバ制御部
- 1 0 0、2 0 0 計算機システム
- 3 0 0 通信路

【図1】



100,200: 計算機システム

300: 通信路

1: 存在情報入力部

2: 存在情報登録部

3: 存在情報テーブル

4: デバッグコマンド入力部

5: ホスト制御部

6,12: デバッグコマンド形式変換部

7,13: デバッグ部

8,14: アプリケーションプログラム

9,15: デバッグ結果形式変換部

10: デバッグ結果出力部

11: サーバ制御部